Multimedia Clinical Simulation based on Patient Records: Authoring, User Interface, Pedagogy

Ramon M. Felciano[1] and Parvati Dev PhD[2]
[1]Section on Medical Informatics, Stanford University School of Medicine
MSOB X-215, Stanford, CA 94305-5479
[2]SUMMIT, Stanford University School of Medicine
MSOB X-329, Stanford, CA 94305-5470

Abstract: As an alternative to available computer-based clinical simulations that mimic patient encounters, we developed a clinical case simulation that more closely resembles the patient record. Our system, implemented in a Macintosh program called Short Rounds, features rapid and customized case authoring by editing a structured text file, a dynamic user interface that presents a case-specific screen layout, and a pedagogical model that is suitable for teaching third-year medical students. We believe this approach allows faculty members to create multimedia case simulations in shorter periods of time than in available clinical simulations.

INTRODUCTION

Computer-based clinical simulations have been used extensively to give medical students experience in medical problem solving, establishing diagnoses, and performing clinical procedures. A wide variety of approaches have been used to model and present patient case information; most require either extensive data entry or formal knowledge modeling. We developed a third system that focuses on presentation and on rapid authoring of clinical materials, and which greatly reduces data entry and eliminates formal knowledge modeling.

BACKGROUND

Computer-based clinical simulations have been difficult and tedious to program. Authoring approaches include manual authoring of each case [6]; manual authoring of the baseline values for a healthy individual with patient variables as a subset of the normal values [8]; and automatic authoring from a knowledge base [7]. Our work is based on the manual approach to authoring.

In all of the previously mentioned authoring approaches, the granularity of the data are the individual symptoms or findings and modifiers of the findings. The advantage of realism is counterbalanced by the tedium of authoring and checking a large number of findings, both normal and pathologic. For the system architect, creating a usable interface for each clinical domain usually means that different versions of the system must be maintained.

From the user's point of view, while some clinical simulations provide a wealth of findings for each case [2, 7], this data-rich format often leads to an awkward user interface. The user is required to scroll through long lists or move through several levels of menus to reach a particular finding. Evidence suggests that this "high fidelity" model, which provides responses to a large number of user queries, may not be best suited to use by third year medical students and that a "pedagogic" model that provides explicit cues on the nature of the case is more appropriate for students just being introduced to clinical medicine. [5]

A second problem in authoring is that the author must usually learn a new computer program that forces him transform his clinical knowledge into a form the computer can use. These authoring tools can be tedious to use because they often force authors to think about the case differently than they are accustomed to doing. Clinical faculty want to make their knowledge available to students but resist modeling it explicitly (as in knowledge-based or expert systems [1]) or presenting it in a fixed format (as in many multimedia presentation systems [3, 4, 8]).

OBJECTIVES

We set out to design a system that addressed some of the limitations of existing computer-based clinical case simulations and their authoring environments. Our target user group was third-year medical students who were just learning clinical medicine; our target author group was the clinical faculty members who taught them. Our goals were to develop a system that provided (1) a pedagogy and scope of clinical information suitable for third year medical students, (2) rapid case authoring in a manner that was natural to the author, and (3) a case-specific user-interface that let students go through cases quickly.

0195-4210/94/$5.00 © 1994 AMIA, Inc.

Pedagogy and Scope of Information

Our central information goal was to focus the student user on the pedagogical points of the case. Clinical simulation programs such as Real Problems and DxR provide a detailed data-driven model of a patient case [8]. The Chest Pain module in Real Problems, for example, has a data set of over 200 items that define a patient case (30 history items, 49 physical findings, etc.) DxR includes a similarly exhaustive set of findings [Irwin, personal communication]. Therefore, these programs have a rigid model of a patient case, where the findings available through the user interface do not vary on a case-by-case basis. Consequently, students face a diffuse body of information and are not guided to the essential insights in the case. In contrast with high fidelity simulations such as DxR, the data granularity in our simulation is a cluster of findings rather than a single finding.

Thus, our system should allow authors to transform the existing clinical information in the patient cases into the user interface of the program, displaying an author-definable selection of findings to the user, and presenting broad rather than deep selection of information. By allowing faculty to integrate existing case materials such as patient records and student handouts, and allowing them to control how this information gets displayed, we hope to encourage the development of a large library of shareable clinical cases.

Case Authoring

We focused on developing a rapid and simple authoring mechanism that would transform the author's clinical information into an interactive presentation for the student. Our goal was to develop a system that would enable the faculty member to spend less than 3 hours to author a single clinical case, including data entry and testing, and that supported the author's personal model of case presentation.

Because obtaining high-quality digital media is difficult (they are hard to locate, and usage rights are complicated [Dev & Felciano, unpublished work]), our system presented information textually, using a graphic interface and page layout techniques, with media as an option, not crucial, feature.

User Interface

The user interface of our system had to present clinical information in a way that communicated the case author's pedagogic goals, and allowed students to review the cases in under 20 minutes per case. Given these constraints, we needed a system that lets faculty control the presentation and flow of information without much formal programming or explicit direction. We did this with a special layout algorithm, described in the next section, that builds the user interface based on the structure of the clinical case.

In order to facilitate access to patient-case data, we decided that all user prompts should be immediately visible on the screen, and not hidden in popup menus or dialog boxes. Furthermore, wherever possible, user interaction is in the form of a single click. We expect the single-click focus of the interface to simplify the porting of the interface to other platforms, especially mobile platforms where interactions such as double-clicking or dragging with a pen may be more difficult to perform. [4]

We also decided that any piece of information would be available within two mouse clicks, regardless of what part of the patient case the user was viewing. This meant the presentation module would need a screen layout that presented only the clinical information relevant to the case, and allowed for single-click navigation between screens.

SYSTEM OVERVIEW

We have attempted to achieve these goals by building a program called Short Rounds. Short

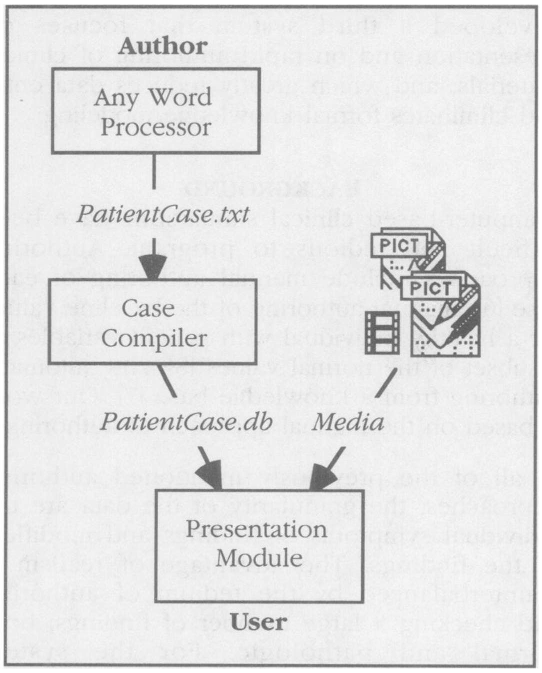


Figure 1. Short Rounds System Design

Rounds consists of a case compiler and a presentation module (Figure 1), both implemented as HyperCard stacks. The case compiler takes an author's text file and compiles it into a database format for the presentation module to use. Compiled cases typically occupy less than 50K of disk space (not counting media files).

The presentation module reads a single compiled case and displays it to the user in a graphic interface. Media files are stored in a separate directory that accompanies the file; they are combined with the textual elements of the case by the presentation module.

User Interface

Short Rounds displays patient information from one of three categories: History and Physical, Tests, and Diagnosis and Treatment. Each screen is separated into two areas: a fixed header bar along the top (A in Figure 2), and a content area underneath. The content area is further separated into an button area on the left (B. in Figure 2) and a findings area on the right (C. in Figure 2).

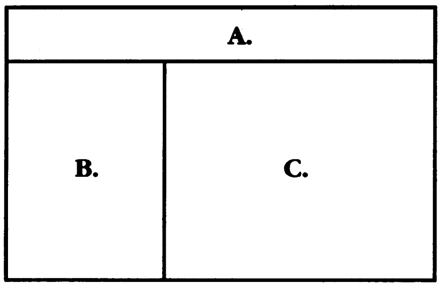


Figure 2. Screen Areas

Each screen displays a single category of information; the button area contains button panels that split this information into subcategories. Figure 3 shows a History and Physical category with three subcategories: History, Physical Exam, and Systems Exam.

The header bar contains the case name and three navigation buttons to move between categories. The header bar also provides buttons to load a new case, access on-line help, and get a basic overview of the program goals. Users click on a category button to move to the screen of their choice.

One or more button panels on the left side of the content area prompt the user to what findings are available. When the user clicks on a button in the panels, text, graphic, and/or sounds relevant to the finding are displayed in the center of the content area.

Finding results that the user has displayed are marked with a bullet (•) in the button panel. In Figure 3, the user has asked about Present Illness, Family/Social History, has done a General, Skin, and HEENT exam, and is currently viewing the patient's Past History.

Once the user has taken a history, performed the physical examination, and ordered tests, he/she can establish a diagnosis and order treatment. Treatment cannot be ordered until a diagnosis has been indicated. The user is given immediate feedback as to whether a correct, incorrect, or possible correct diagnosis was chosen. A summary button that displays a case write-up is available once the user is finished.

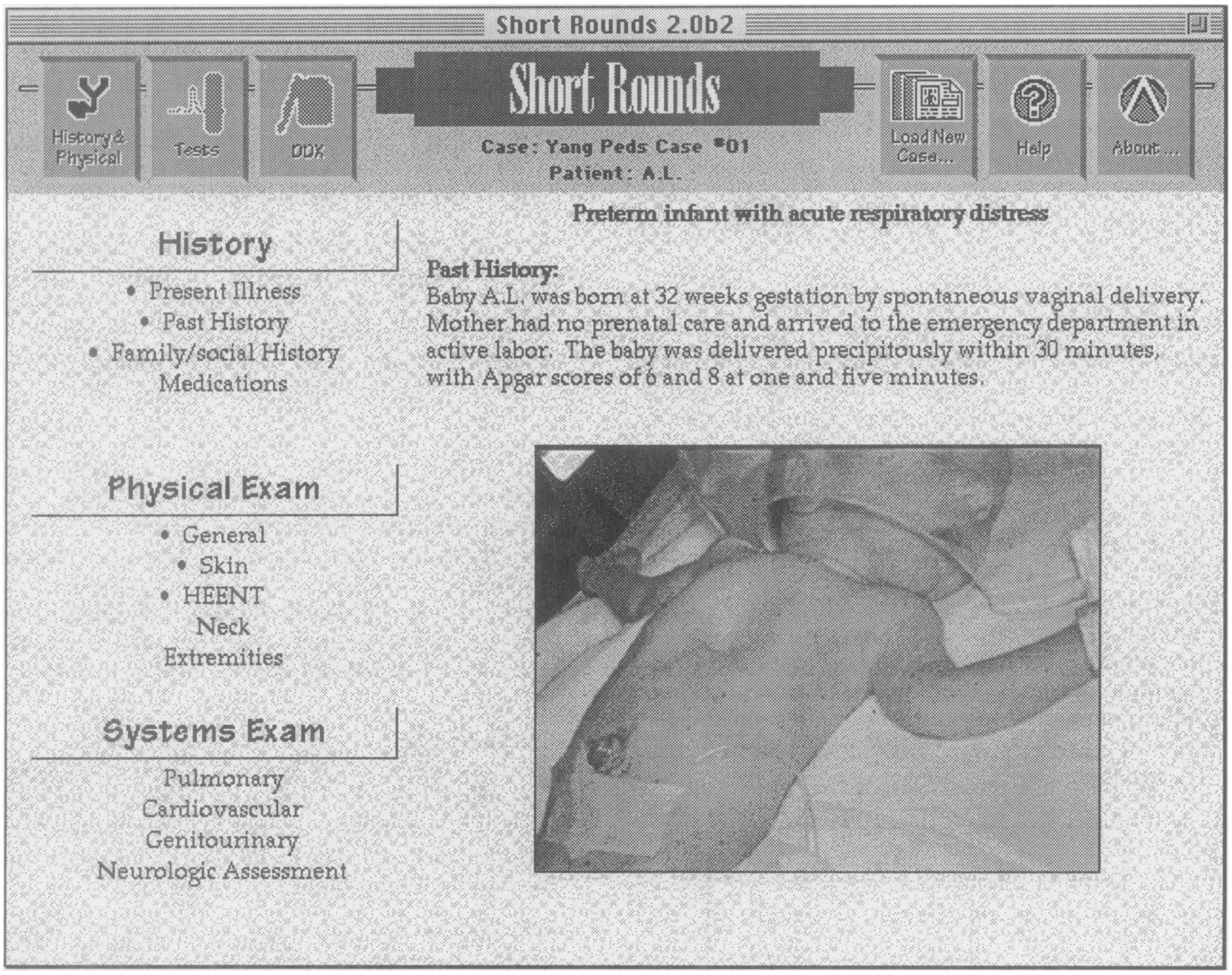
Authoring

Authors create Short Rounds cases by editing a structured text file using any word processor on any platform. This allows authors to use text editing software they are familiar with, and obviates training them on a new tool.

The text file contains natural language prompts that are parsed by the Short Rounds program. Special delimiters indicate the start of each cluster of clinical findings. Each such cluster contains text that may be entered in the patient record as well as file names for media that amplify the text, such as the x-ray image, the patient's heart sound, or a digital movie. The media elements are optional; cases can be entirely textual.

The specific guidelines for developing a case are purposefully simple. Double-bullets (••) mark the names of Categories and Subcategories. Single bullets (•) indicate clinical findings. Special prompts, such as "Picture:", are used to point to media file names. The resulting text file is both human and machine readable, and, most importantly, resembles a conventional case write-up.

Figure 4 shows an excerpt defining entries in the "History" sub-category of the "History & Physical" screen.

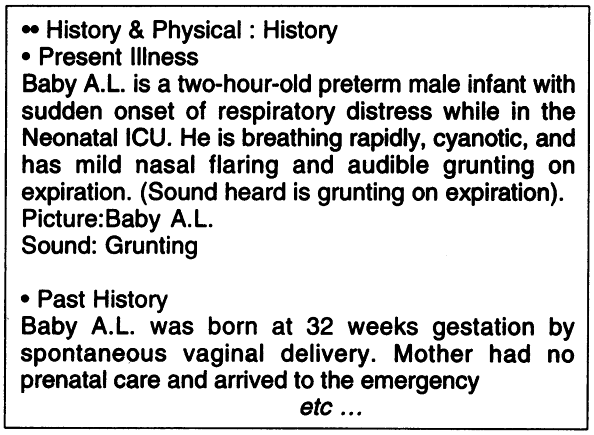


Case: Yang Peds Case #01
Patient: A.L.

Preterm infant with acute respiratory distress

Past History:
Baby A.L. was born at 32 weeks gestation by spontaneous vaginal delivery. Mother had no prenatal care and arrived to the emergency department in active labor. The baby was delivered precipitously within 30 minutes, with Apgar scores of 6 and 8 at one and five minutes.

Figure 3. Example of a History and Physical screen



•• History & Physical : History
• Present Illness
Baby A.L. is a two-hour-old preterm male infant with sudden onset of respiratory distress while in the Neonatal ICU. He is breathing rapidly, cyanotic, and has mild nasal flaring and audible grunting on expiration. (Sound heard is grunting on expiration).
Picture:Baby A.L.
Sound: Grunting

• Past History
Baby A.L. was born at 32 weeks gestation by spontaneous vaginal delivery. Mother had no prenatal care and arrived to the emergency
etc ...

Figure 4. Sample Case File

Case Compiler

The case compiler verifies the basic structural integrity of the file, and converts it into a more compact database format. Because there is no data or format verification done by the word processor, the compiler attempts to strip out superfluous data such as trailing spaces, and ignores some common formatting errors (for example, using "*Graphic:" as a prompt instead of "•Picture:").

Authors usually save in a compiled format for the presentation module to load. During the authoring stage, however, the compiler can be integrated with the presentation module. This allows authors to load an un-compiled case into the presentation module and have it compiled just before it is displayed. Since this compilation can take as long as two minutes for large cases, cases are typically distributed in compiled form once they are complete.

Presentation Module

The presentation module dynamically constructs the screen layout of the content area from the structure of the patient case file. This process involves (1) linking the user interface elements to clinical content and (2) positioning the user interface elements on the screen. Loading a compiled case and preparing the screens typically takes less than 7 seconds.

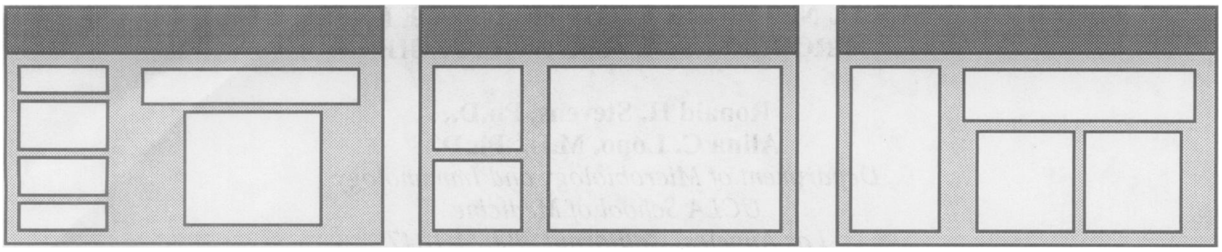


Figure 5. Alternate Layouts

The layout algorithm uses the number of finding subcategories, the number of media elements, and the types of media to build the layout. During a first pass, the program parses each cluster of findings in the patient case, creates a button on the fly for that cluster, and links text and pertinent media (images, video clips, or sounds) to the button. During a second pass, the program organizes all the buttons in an aesthetic layout in the window. Each panel of findings and result display is a rectangular panel that can be moved or resized on the screen. The layout algorithm preserves a consistent graphic design by maintaining proportional sizing and spacing on the screen. Representative layouts (Figure 5) show how panels could be arranged to display different combinations of subcategories and findings.

Since all findings are listed along the left side of the screen and the navigation buttons are preserved in the header bar, users can get to any finding by clicking no more than twice: once on the category and once on the finding.

USAGE

Short Rounds is currently in the development stages, and has been used to develop six neonatal intensive care cases. During this initial development process, we have found it easy to enter the clinical information such as history and physical findings. Informal discussions with faculty have indicated a need for Short Rounds cases to include an explicit teaching element in addition to the clinical information; this teaching element would allow faculty to explain why this is a good teaching case.

ACKNOWLEDGMENTS

The authors thank Michael Walker, Tom Rindfleisch, and Philip Constantinou for guidance and support. Lawrence Mathers, MD, provided the initial set of clinical cases. Nora Sweeny provided valuable editorial comments on earlier drafts of this paper. This work was sponsored by the Stanford University Medical Media and Information Technologies (SUMMIT) laboratory, and partially supported by a gift to SUMMIT from Dr. Peter Bing and Mrs. Helen Bing. Some computing services were provided by the CAMIS resource at Stanford, NIH grant No. LM-05305.

REFERENCES

[1] Chin, H.L. and Cooper, G.F., Case-based tutoring from a medical knowledge base. *Comput Methods Programs Biomed*, 1989. 30(2-3): p. 185-98.

[2] Cundick, R., *et al.* ILIAD as a patient case simulator to teach medical problem solving. in *13th Annual Symposium on Computer Applications in Medical Care*. 1989. Washington, DC: American Medical Informatics Association.

[3] Dichter, M.S., Greenes, R.A., and Bergeron, B.P. Authoring multimedia clinical problem-solving exercises with CaseBase. in *Proc Annu Symp Comput Appl Med Care*. 1991.

[4] Felciano, R.M., Daane, S.P., and Dev, P. Clinical Pearls and Short Rounds: Two Shells for Multimedia Case Presentations. in *AMIA Spring Congress*. 1994. San Francisco, CA: AMIA.

[5] Friedman, C.P., France, C.L., and Drossman, D.D., A randomized comparison of alternative formats for clinical simulations. *Med Decis Making*, 1991. 11(4): p. 265-72.

[6] Lyon, H.J., *et al.*, PlanAlyzer, an Interactive Computer-assisted Program to Teach Clinical Problem Solving in Diagnosing Anemia and Coronary Artery Disease. *Academic Medicine*, 1992. 67(12): p. 821-828.

[7] Miller, R.A., *et al.*, The INTERNIST-1/QUICK MEDICAL REFERENCE Project—Status Report. *The Western Journal of Medicine*, 1986. 145 (6): p. 816-22.

[8] Perper, E.J., Felciano, R.M., and Dev, P. Real Problems: A Layered Approach to Constructing a Patient Simulation. in *17th Annual Symposium on Computer Applications in Medical Care*. 1993. Baltimore, MD: American Medical Informatics Association.